REMARKS

Claims 4, 5, 7, 8, 11, 12, 15, 16, 19, 20, 21, 25, 41, 43, 45, 48, 49, 51, 52, and 56-62 are pending in the application.

By the foregoing Amendment, claims 4, 5, 7, 8, 11, 12, 15, 16, 19, 20, 25, 41, 43, 45, 48, 49, 51, and 52 are amended. Claims 1-3, 6, 9, 10, 13-14, 17-18, 22-24, 26-40, 42, 44, 46-47, 50, and 53-55 are canceled without prejudice or disclaimer (of which claims 2, 13, 17, 27, 38, 42, and 53 were canceled in the Response submitted January 9, 2008). New claims 56-59 are added. The specification also is amended.

Claims 7 and 12 are amended to incorporate limitations from their base and intervening claims, as well as additional limitations described in the application. Claim 16 is amended to define components of the text pattern recognition rule. Claim 20 is amended to incorporate limitations from its dependent claims 22-24, as well as additional limitations described in the application. Claim 41 is amended to incorporate limitations from its dependent claims 46 and 47, and to more precisely define the annotating and extracting steps. The remaining dependent claims are amended as necessary for consistency with their base claims.

New claims 57, 58, and 59 depend respectively from claims 7, 12, and 41 and are directed to the feature of the pattern recognition language that annotated text is represented as a single view of a document expressed as inline XML, as described in amended paragraphs 00011, 000115, 000175, and 000186 of the application.

New claims 60, 61, and 62 depend respectively from claims 7, 12, and 41 and are directed to the feature of the pattern recognition language that uses XPath for traversing XML-based tree representations in the annotated text.

Paragraphs 00011, 000115, 000175, and 000186 of the application are amended to state explicitly the feature of the invention that annotated text is represented as a single view of a document expressed as inline XML. A person of ordinary skill in the art would recognize this feature to be an inherent characteristic of the invention from reading the description of the invention as described in the application as originally filed. If the Examiner considers it necessary to do so, Applicant will provide a statement from inventor Mark Wasson to this effect.

These changes are believed not to introduce new matter, and entry of the Amendment is respectfully requested.

Based on the above Amendment and the following Remarks, Applicant respectfully requests that the Examiner reconsider all outstanding objections and rejections, and withdraw them.

Objections to the Specification

In paragraph 5, the title was objected to as not being clearly indicative of the invention to which the claims are directed. This objection is believed to be overcome by the foregoing amendment to the title. If the Examiner finds that the amended title is still not sufficiently descriptive, the Examiner is invited to suggest an alternative.

In paragraph 6, the specification was objected to as not providing proper antecedent basis for a computer program product as claimed. This objection is overcome by the cancellation of the claims directed to a computer program product.

Rejection under 35 U.S.C. § 101

In paragraph 7 of the Office Action, claims 26-40 were rejected as being directed towards non-statutory subject matter under section 101. Claims 27 and 38 were canceled in response to the previous Office Action. This rejection is overcome by the cancellation of claims 26, 28-37, and 40.

Rejection under 35 U.S.C. § 112, ¶ 1

In paragraph 9 of the Office Action, claim 16 was rejected under section 112, first paragraph, for lack of enablement for the limitation of a "text pattern recognition rule" and also as being a "single means claim" for which the disclosure is non-enabling, citing section 2164.08(a) of the MPEP. This rejection is believed to be overcome by the foregoing amendments to claim 16.

Rejections under 35 U.S.C. § 103

In paragraph 10 of the Office Action, claims 1, 3, 4, 8, 9, 10, 12, 14-16, 18, 19, 25, 26, 28, 29, 33-35, 37, 39, 40, 41, 43, 44, 48-50, 52, 54, and 55 were rejected under section 103(a) as being unpatentable over Cunningham et al. ("Developing Language Processing Component with GATE (a User Guide)," 2001 -2002) in view of Tokuda et al. (US 2003101 54070), and further in view of Feldman et al. (US 6,442,545); in paragraph 11, claims 20 and 21 were rejected under section 103 as being unpatentable over Cunningham et al. in view of Tokuda et al.; in paragraph 12, claims 5-7, 30-32, and 45-47 were rejected under section 103 as being unpatentable over Cunnigham et al in view of Tokuda and Feldman, as applied to claim 1, and further in view of Broder et al. (US 200410243645); in paragraph 13, claims 11, 36, and 51 were rejected under section 103 as being unpatentable over Cunningham et al. in view of Tokuda and Feldman, as applied to claim 10, and further in view of Marcus et al. ("The PENN Treebank Annotating Predicate Argument Structure," 1994); and in paragraph 14, claims 22-24 were rejected under section 103 as being unpatentable over Cunnigham et al. in view of Tokuda, as applied to claim 1, and further in view of Broder et al.

The present invention is directed to a fact extraction system (also known as an information extraction system) that first annotates the text in some document with any number of attributes, including linguistics, semantic and orthographic attributes, and then provides users with a pattern recognition language in which users write rules that look for patterns of attributes in order to identify, extract, and label the corresponding text in that document. A pattern might looks for the direct object of "John shot" in order to identify and extract the victim/target. The inventors have used the tool to determine which people in some text are attorneys and to associate those attorneys with the correct law firms also found in that text.

The inventors chose to use XML to represent the annotated text because it supports both the annotation of text strings and also the representation of syntactic structure, structure that can be produced through a linguistic parser adapted to annotate the text. However, the decision to use XML posed two major challenges to the inventors, each of which is the basis of significant features of the invention, and claim limitations corresponding thereto.

- The inventors wanted to be able to exploit both the annotated string and the underlying parse trees in their fact extraction applications, so they created a pattern recognition language that combines regular expression and tree traversal capabilities, using XPath for the tree traversal work.

- XML requires well-formedness, but independently developed annotators can produce annotation bracket pairs that cross one another. Crossed bracket pairs are not permitted in well-formed XML. The inventors therefore created a method that uncrosses crossed bracket pairs in a way that does not prevent accessing the text as originally marked.

The Office Action asserts that it would have been obvious to combine regular expression capabilities and tree traversal capabilities, using XPath for the tree traversal work, in a pattern recognition language, based on the teachings of Cunningham et al in view of Tokuda et al., and further in view of Feldman et al.

In combining the teachings of Cunningham et al., Tokuda et al., and Feldman et al., the Office Action concludes: "It would have been obvious to one of ordinary skilled in the art at the time the invention was made to have modified the fact extraction taught by Cunningham et al. in view of Tokuda et al. with the use of XPath functionality as taught by Feldman et al. ..."

The first significant feature of the invention is the pattern recognition language. Most rule-based fact extraction systems use some variant of regular expression-based pattern matching as the basis for their pattern recognition language; others focus on parse tree traversal. The inventors of the present invention devised a system that combines both by integrating XPath tree traversal capabilities with regular expression-based pattern matching.

Cunningham et al is a user manual for the University of Sheffield's GATE system, a rule-based fact extraction system; and is discussed in the Information Disclosure Statement submitted March 10, 2004. As conceded in the Office Action, Cunningham et al. "does not specifically teach annotation of tree-based attributes". Applicants are in general agreement with the Office Action's characterization of Cunningham et al.

Tokuda et al. discloses a system that helps language learners through parser-based grammar checking and correcting; it uses a parser to support the grammar checking, and the parser results are stored in a parse tree representation (annotation) of the text. The Office Action asserts that "[i]t would have been obvious to one of ordinary skilled in the art at the time the invention was made to have modified the fact extraction as taught by Cunningham et al. with the annotation of tree-based attributes as taught by Tokuda et al. for the purpose of accurately analyzing natural language sentence," but concedes that "Cunningham et al. in view of Tokuda et al. do not specifically disclose the XPath-based functionality."

Feldman et al. discloses a data visualization system that uses the output of an information extraction process combined with information pulled from a tree-based taxonomy. As noted in the Office Action, "Feldman et al. does disclose the use of XPath-based [tree traversal]" for navigating the hierarchical (tree-based) taxonomy.

With respect to Tokuda et al., language parsers do indeed produce parse tree representations of text, so it is not surprising that Tokuda et al. represents its parser-based annotation in a tree format. However, Tokuda et al. is not a fact extraction application, and it provides no insight at all into how to navigate parse trees for fact extraction purposes, whether through the use of XPath or any other means.

The University of Sheffield, where Cunningham et al. and GATE are based, has a strong program in computational linguistics, and parse trees are widely understood in the field of computational linguistics as a means for representing parser-based annotations of text. The problem has been how to effectively combine surface based annotations[1] that regular expressions

[1] The term "surface-based annotations" is not used in the specification. "Surface-based annotations" are annotations assigned to tokens or to sequences of tokens. The tokens generally represent text strings, such as a word, a name or white space, and the text strings are what the reader sees, i.e., the surface strings.

can exploit with tree-based annotations that tree navigation functionality (such as XPath) can exploit.

Because Tokuda et al is not a fact extraction system, and because it does not provide any insight into how to navigate parse trees for fact extraction purposes, reading Tokuda et al. would not provide anyone skilled in the art with information needed to combine Cunningham et al with Tokuda et al in a way that would support enhanced fact extraction capabilities.

Feldman et al. discloses a data visualization application that uses XPath for navigating a taxonomy, a hierarchical organization of terms and term categories, somewhat like a thesaurus. Feldman et al. does not use XPath as part of an information extraction process, and thus it does not provide any insight into how XPath could be incorporated into an information extraction system or pattern recognition language. Furthermore, the Office Action cites Feldman et al. as teaching (in the Abstract and col. 12, lines 10-25) "where POS [part of speech] is taken into consideration to determine relationships." These relationships are not identified through traditional fact extraction means, but rather through their data visualization process, i.e., the browser. Feldman et al. Figure 6, for example, shows how data visualization is done, where names and other terms extracted from a set of one or more documents, perhaps normalized with the help of the taxonomy, are organized in a circle – lines indicate identified co-occurrence

Paragraph 0006 of the application defines "annotations" as "attributes, or values, assigned to words or word groups that provide interesting information about the word or words. Example annotations include part-of-speech, noun phrases, morphological root, named entities (such as Corporation, Person, Organization, Place, Citation), and embedded numerics (such as Time, Date, Monetary Amount)." Paragraph 0008 defines "attributes" as "features, values, properties or links that are assigned to individual base tokens, sequences of base tokens or related but not necessarily adjacent base tokens (i.e., patterns of base tokens). Attributes may be assigned to the tokenized text through one or more processes that apply to the tokenized text or to the raw text." Paragraph 00010 defines "base tokens" as "minimal meaningful units, such as alphabetic strings (words), punctuation symbols, numbers, and so on, into which a text is divided by tokenization. Base tokens are the minimum building blocks for a text processing system"; while paragraph 00028 defines "tokens" as "a minimal meaningful unit, such as an alphabetic string (word), space, punctuation symbol, number, and so on."

Thus, "surface-based annotations" is simply a short-hand term for they types of annotations described in the present application.

relationships, and line thickness is an indicator of the strength of the relationship, based in part on how often those terms co-occur across a set of documents. Data visualization is used to identify the relationships, not fact extraction.

Tokuda et al. shows a valid use of parse trees in a grammar checking application, but that application is not related to fact extraction. Feldman et al. shows a valid use of XPath for hierarchy navigation. Although Feldman et al.'s application uses the output of a fact extraction system combined with taxonomy information to support data visualization, Feldman et al. does not use XPath in a rule-based fact extraction system. Because grammar checking applications and taxonomy navigation applications are not related to fact extraction, and because none of the cited references use tree navigation to support fact extraction, it is respectfully submitted that Cunningham et al. in combination with Tokuda et al. and Feldman et al.[2] would not have made it obvious to one of ordinary skilled in the art how to combine regular expression-based pattern recognition and tree navigation-based pattern recognition into a unified fact extraction pattern recognition language.

Claims 7, 12, and 41 all reflect the combination of both regular expression-based pattern matching and tree traversal pattern matching in annotated text, where XML is used as a basis for representing the annotated text.

Another significant feature of the invention deals with well-formed XML and annotators that do not align. In the Office Action, Broder was cited in paragraphs 12 and 14 as teaching "the use of independent annotators (see [0153]) (e.g. it is seen that independent annotations are

[2] Although not relevant to the obviousness argument because it antedates the filing date of the present application, it is noted that even today, the ongoing GATE project (http://www.gate.ac.uk/) still does not include XPath tree navigation. Also, ClearForest (Feldman et al.) has since developed its own rule-based pattern recognition language for fact extraction (DIAL4 – DIAL4 Language Reference Manual, Version 5.0, March 2004; DIAL4 Language Reference Manual, Version 6.1 SP2, June 2005), but DIAL4 does not include tree navigation capabilities and XPath-based functionality.

used for each type of word pairs."; and it was stated that "[t]he motivation to have used independent annotators [as taught by Broder in the fact extraction and annotation taught by Cunningham *et al.* in view of Tokuda in view of Feldman] is to resolve the issue of overlapping annotations that occurs in nested XML (see Broder *et al.* [0128])."

XML markup must be well-formed, i.e., bracketed components may be nested in one another, but bracket pairs may not overlap, or cross boundaries, with one another. When several components of some system are developed to work together to annotate some text, bracket overlap is generally not a problem, because one annotator generally was designed to work off the output of other annotators.

When independently developed annotators are applied to the same text, however, overlap, or crossed boundaries, often occurs because the individual annotators are ignorant of the outputs of the other annotators. This is further complicated because no text annotation system is perfect; annotation errors often lead to overlap, and thus to non-well-formed XML. The fact extractions system in accordance with the present invention enables the integration of independently developed annotators into applications as needed.

There are other approaches to annotating text that do not have XML's well-formedness requirement, but because the inventors wanted to use XML-related functionality such as XPath, they needed to represent annotated text with a single well-formed XML-based structure.

Both Broder and the present invention undertake to solve the problem of applying multiple, independent, and minimally-coordinated annotators to a document. In its preferred embodiments, Broder eschews inline XML annotation in order to avoid overlapping annotations, relying instead upon stand-off markup indexed to document offsets stored in an inverted file system. In contrast, the present invention embraces XML as a method for storing semantic annotations inline with the serial, grammatical, and rhetorical context of the document. In order to preserve the serial, grammatical, and rhetorical context of a document that has been multiply-annotated, the present invention employs a novel and non-obvious method of "uncrossing" overlapping annotations so that XML well-formedness constraints are not violated. Retaining XML as a storage medium leverages the established value of XML. Preserving the serial, grammatical, and rhetorical context in inline XML then enables simplified methods of writing extraction rules and applying said rules to the annotated documents.

In the abstract, Broder states that "... Operating the at least one text analysis engine generates a plurality of views of a document, where each of the plurality of views are derived from a different tokenization of the document." In paragraph 00154, Broder states that "preferably there is at least one inverted file system for storing tokens (see FIG. 15), and at least one inverted file system for storing, for each of the views, the annotations, a list comprising occurrences of respective annotations and, for each listed occurrence of a respective annotation, a set comprised of a plurality of token locations, where a given token location may be spanned by at least one annotation (see FIG. 13)."

In contrast, in the present invention as recited in new dependent claims 57-59, a plurality of text analysis engines (or "annotators") can generate a *single* view of a document (expressed as *inline* XML) based upon a *single* tokenization of the document.

Although Broder tolerates XML representation in places (e.g. at 0121), Broder generally regards XML as a problematic and non-preferred technology. In particular, at paragraph 0128 Broder states:

> Further in consideration of XML, in some embodiments a disadvantage of the XML representation is that a TAE 130 may produce overlapping annotations. In other words, annotations are not properly nested. However, XML would not naturally

represent overlapping annotations, and further mechanisms may be employed to provide a solution.

In the present invention, a single XML view can be a preferred storage medium by employing a method by which logically overlapping elements (referred to by Broder in paragraph 0128 as "cross-over spans"); "crossed nodes" or "crossed boundaries" (Wasson 0125 *et passim*) can be "uncrossed" so as to preserve XML well-formedness.

The "further mechanisms" Broder teaches to provide a solution to overlapping annotations (cf. paragraph 0129 *et seq.*) do not include the "uncross" mechanism taught in the present application at paragraph 0125 *et seq.* and recited in the claims.

Broder discloses annotations that are "uncrossed" in virtue of being stored in "an inverted file data structure" which implements the long-known and unoriginal method of stand-off markup. The annotations of the present invention, by contrast and by virtue of the present invention's uncross method, can be stored in a single, inline XML document. As a result, semantic annotation is retained inline with the serial, grammatical, and rhetorical structure of a document throughout processing. Being XML-compliant at all stages of processing simplifies the maintenance of the entire system of documents and annotations. The ability to store all semantic annotations in their natural inline serial, grammatical, and rhetorical context then greatly simplifies the writing and debugging of extraction rules.

With respect to the claim limitations relating to independent annotators, it is respectfully submitted that the Office Action reflects a misunderstanding of the function of independent annotators. It is *not*, as stated in the Office Action, "to resolve the issue of overlapping annotations." Rather, independent (i.e., different) annotators are used, each for its special expertise (see, for example, claims 5 and 22, and application paragraph 00021). For example, a Hindi Named Entity Annotator may recognize that "Narasimha" is a common Tegalu given name, thus helping a subsequent English annotator parse an English sentence containing that name. The use of multiple annotators is necessitated by the complexity of the annotation task. Contrary to the assertion in the Office Action, multiple annotators do not *resolve* the issue of overlapping annotations; they *create* it.

Marcus et al. was cited in the Office Action as teaching the identification of non-contiguous attributes. It is noted that well-formed XML, such as results in the present invention from resolving conflicting annotation boundaries, has no problem encoding "non-contiguous" nodes. It is *overlapping* nodes (i.e., conflicting annotation boundaries) that are at issue, and that are resolved in the present invention by resolving the conflicting annotation boundaries.

It is respectfully submitted that Broder does not supply the teaching or the motivation as set forth in paragraphs 12 and 14 of the Office Action. By either using something other than XML or by storing conflicting annotations in separate representations, Broder sacrifices the advantages of having a single well-formed representation of annotated text, including the ability to apply an ever expanding number of XML-based applications and functionality to that content. Broder and Wasson end up at very different points with respect to their ability to use those applications. Again, being XML-compliant at all stages of processing also simplifies the maintenance of the entire system of documents and annotations.

In both paragraphs 12 and 14, the Office Action states that "[t]he motivation to have used independent annotators is to resolve the issue of overlapping annotations that occurs in nested XML." Actually, the motive for creating a process that resolves the issue of overlapping annotations that occur in XML is to make it possible to use independent annotators whose results may overlap. The approach of the present invention allows both the use of a growing number of

XML-based applications with the annotated text and the incorporation of independent annotators in the fact extraction process, a combination that Broder's approach does not permit.

In summary, with respect to the integration of tree-based functionality into a regular expression-based fact extraction pattern recognition language, it is respectfully submitted that neither Tokuda et al nor Feldman et al is a fact extraction-based application; and thus Cunningham et al. in view of Tokuda et al nor Feldman et al do not provide those skilled in the art with any insight into how to integrate tree traversal in general, and XPath in particular, into a regular expression-based pattern recognition language.

With respect to the invention's approach to addressing the overlapping annotations problem, Broder et al does not produce a single XML-based representation of annotations, whereas the invention does, giving the invention's approach advantages in system support and maintenance, as well as the ability to integrate the invention with third party XML-based technologies.

In view of the foregoing, it is respectfully submitted that the invention as recited in independent claims 4, 5, 7, 8, 11, 12, 15, 16, 19, 20, 21, 25, 41, 43, 45, 48, 49, 51, 52, and 56 is patentable over the cited prior art; and that the rejections should be withdrawn.

Conclusion

All objections and rejections have been complied with, properly traversed, or rendered moot. Thus, it now appears that the application is in condition for allowance. Should any questions arise, the Examiner is invited to call the undersigned representative so that this case may receive an early Notice of Allowance.

Favorable consideration and allowance are earnestly solicited.

Respectfully submitted,

JACOBSON HOLMAN PLLC

Date: August 26, 2008

By:

Allen S. Melser
Registration No. 27,215

Customer No. 00,136
400 Seventh Street, N.W.
Washington, D.C. 20004
(202) 638-6666

Enclosures: Petition For Extension Of Time
Credit card payment form